Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/6/20230816

The application of probabilistic programming in statistics

Yi Zhang

Department of Statistics, University of California, Davis, Davis, United States

johzhang@ucdavis.edu

Abstract. This article focuses on three aspects of Probabilistic reasoning systems for applications, including Predict future events, Infer the cause of events, and learn from past events to better predict future events. In addition, the article explains that the data will become more accurate as the number of samples increases, and finally, the article describes the application of Turing-Completed Systems in statistics, a kind of probabilistic programming system. This article starts with a soccer competition as an example, then will explain the concepts in depth behind the example.

Keywords: Probalistic Programming, Statistics.

1. Introduction

Have readers ever thought that when do something, normally, people will first evaluate its outcome, such as predicting a good outcome, a normal outcome, and a terrible outcome, and then consider what is the next step after each outcome. Or maybe people don't know how's these things will happen, thus afraid to work on the problem, instead, just sitting at the table and trance. But now, probabilistic programming can help solving these problems. Probabilistic programming is a way to create systems that helps to make decisions when facing uncertainty. Probabilistic programming is everywhere in people's daily lives. However, people can't find them easily. And, there are three specific kinds of reasoning that probabilistic reasoning systems can do. In the world, all problems are uncertain until the outcome is known. For example, when a product is introduced to the market, there is no guarantee that it will succeed. Such as the quality of competitors' products, economic crisis, market demand. In other words, it is impossible for a person go through the result 100% after making the decision. That's the reasons when making decisions, people use the language of probability. The language of probability can help making decision, although people still can't make sure how the result will be. But it estimates the probability of success of that outcome, and the probability of failure. For example: If it succeeds, how much profits will the product bring in. If it loses, how much it will lose. These allow people having estimations when making decisions. Another example in sport activity, the statistics show that 9% of corner kicks result in a goal. The current situation is that the attacking team's leader is taking a corner, and the defending team, As the defending back, has a goalkeeper who has just had been injured, and the new goalkeeper is a substitute, He/she has to face a fierce attack from the opposite side. In addition, the weather is also a factor. In conclusion, with adding the factors of the weather(wind), the features of attacker and defender through probabilistic reasoning, the success of this corner kick is 20%. In addition

DOI: 10.54254/2755-2721/6/20230816

to business and sports, Probabilistic reasoning can be everywhere in people's daily lives, it gives people reference when making decisions.

This article will mainly focus on some concepts about probabilistic programming. Like the three ways where Probabilistic reasoning systems can reason, including Predict future events, Infer the cause of events, and learn from past events to better predict future events. As mentioned before. The weather, how strong the attacker is, and the defender's experience. Thus, people can make assumption(prediction) through the knowledge. To infer the cause of events, for example, this powerful player scored a goal in front of the last equally powerful goalkeeper, which is also evidence. But that's not enough. The goalkeeper's information is just as important. Suppose the model works out that the probability of making a goal is 20, the probability of not making a goal is 50, and the average is 30. the first reasoning pattern describes reasoning forward in time, predicting future events based on what people know about the current situation, whereas the second reasoning pattern describes reasoning backward in time, inferring past conditions based on current outcomes. The last one, to learn from past events to better predict future events. For example, after that goal, the morale of the attacking team will be improved. Similarly, the morale of the defending team will also be affected, and the subsequent progress will be affected. Also, the article will explain that, with more data (sample) shown a probabilistic reasoning system will be more accurate. Finally, there will be an introduction of Turing-complete programming, The protagonist of the random process, which are common in people's daily lives. A soccer game is just an example to help people feel interested in it, the article will focus on the three ways in probabilistic reasoning, the difference between the initial model and the final model, and the Turing-complete system to help with the calculation.

2. Probabilistic reasoning systems and turing complete

At many kinds of sport events, betting is everywhere. People speculate on the ultimate winner at the expense of money. In all the bets, the highest percentage of people are glad to put their money to the best teams, which is reasonable, they realize the team is doing well and they are confident that the team they are betting on will be the champion at the end of the tournament. Like the goaling mentioned above, people will determine that will happen in a while based on the information of the team itself and the current situation. Of course, the team with the lowest odds will never have a 100% chance of winning the tournament. In programming, computers will help predict where things will go. Ai might be familiar with the League of Legends (LOL) tournament viewers. Based on their current circumstances and general competence, Ai can analyze the victory percentages of the two teams during the game. In a past study, Riot Games invited team members to answer questions on their feelings and perceptions of the group in order to assess the CI, game performance, and team dynamics of a league team. After the research was finished, Riot Games gave the team's in-game data, particularly recent conditions. This data is secret, of course, for reasons of privacy. The model will automatically assign the squad a suitable score after gathering the data [1]. Of course, there are factors. For example, during a game, a team with a lower score will receive greater instant rating [1] if they take advantage of the team with a higher score. Although the information provided by the show is not entirely accurate, it serves as a general audience reference. Additionally, several sports use unique models. Discrete-time models are typically employed for slower-paced activities like pool, golf, baseball, and shooting. After play of a certain number of holes, balls, shots, and rallies, they will provide sequences of observations that correspond to scores (end of round). Instead, the model can produce sequences of observations for time-critical games like football, hockey, and athletics based on goals scored, changes of possession, changes of lead, and baskets scored after uncountable periods of time [2]. Again, the probability provided by program is not accurate, while they predict the outcome based on the previous conditions and current situation. However, while they can only do is to predict, rather than to control the outcome of the games. On the contrary, the game might become boring without variables.

For most gamblers, their perspectives are always blind, they can't make reasonable inferences from gambling, they can only rely on their own luck. while probabilistic reasoning systems are not, with more perspectives, they are able to infer the cause of events. Bayes' rule, also known as probabilistic

DOI: 10.54254/2755-2721/6/20230816

population coding, is typically used by neural populations to encode probability distributions over stimuli. This kind of coding allows for the implementation of the optimal cue combination in a physiologically plausible network based on the common cause hypothesis. Surprisingly, this merely calls for straightforward linear manipulations on brain activity. The superior colliculus is one example of an area where this implementation makes critical use of the structure of neural variability and yields physiological predictions for activity. Complete integration differs from causal inference in various ways, therefore computational processes for the latter are intended to have a neurological underpinning that generates these linear operations on population activities. An optimum causal inference neural implementation will be an significant step towards a complete neural theory of multisensory perception [3]. Another point of inference is to make more assumptions, such as Logistic Regression. To see what will happen if this assumption is finally established, and what will happen if it is not? For example, in Dota 2, a MOBA similar to LOL, the authors use features of heroes picked and their win rates in DOTA 2 competitions and then predict the outcome using Logistic Regression and Random Forests, in this case, they got 73% accuracy. Similar algorithms are used to predict the outcome in League of Legends with different time intervals, and they got 75% of accuracy. There are some other works tries to predict the outcome of DOTA 2 games based on the heroes picked, but this case uses the Naive Bayes classifier and obtains an accuracy of 58.99%, which is far behind the previous one. In Hanke et al., a hero recommendation system using association rules was developed for DOTA 2 and a neural network was created to predict the result of matches based on the heroes picked. The neural network obtained an accuracy of 88.63% in the test set. Neural networks were also used for predicting results in other sports, like basketball. For instance, they won 73% of accuracy in predicting NBA tournament [4]. Everything has its cause and effect. Collecting more evidence to make the view be more convincing. It is not enough to predict the result from the given information. A good program needs to understand what is causing of these events. And are they being resolved, and what will the outcomes be like if resolving them or not.

Furthermore. the things that happened can also affect each other, just like butterfly effects. When something happens, what happened before can be reasonable, and similarly, or on the other side, what is happening will also affect subsequent behaviors. To simplify, things are related to each other. Everything can be alternative, after occurring of an event, depending on the current situation, the possibility of it reoccurring will change, it might increase or decrease, but it won't stay constant. In addition, this view may not likely to be detected by programming, because computers cannot simulate people's motivation, emotional changes. Researchers have discovered some evidence for a durability bias for negative outcomes in previous studies, even for negative events that people had experienced before and could have learned from (e.g., a loss by one's favorite football team; Gilbert et al., 1998; Wilson et al., 2000). One study explained that people's emotional responses to predicting negative events are more accurate than positive events, but later another study have rejected this view. Second, prior research has not demonstrated a relationship between expected future unfavorable events and a person's mood [5]. In summary, computers are more about simulating people's reactions for reference. Although people's inner thoughts are affected by the situation, the computer cannot fully predict the actual situation.

Slightly different from inference algorithms, the goal of learning algorithms is to generate a new model rather than answer queries. The learning algorithm starts from the original model and updates it according to experience to generate a new model. In the future, the new model could be used to answer queries. Presumably, the answers produced when using the new model will be better informed than when using the original model. With more data (sample) shown a probabilistic reasoning system will be more accurate. The quality of the predictions depends on two things: the degree to which the original model accurately reflects real-world situations, and the amount of data sample. Since probabilities are derived from observations that are determined from samples, confidence intervals are important in statistics. Statistics gathered from samples that are chosen in an objective manner may not reflect the underlying rate in the population from which they were taken. As long as the samples are not systematically biased, more samples will typically yield more accurate answers. For instance, a coin should have a 50% chance

DOI: 10.54254/2755-2721/6/20230816

of being either head or tail. It will be especially biased toward either head or tail at the start of tossing. But as the number of samples rises, the likelihood of tossing will eventually equal 50% for both sides. [6] Tossing a coin is not an accidental event, as more data become available, the original model becomes less significant. For example, the corner ball's success percentage in the aforementioned scenario was 20% in the original model (with wind), but following the interference of various data, the success rate rapidly decreased. To apply the Jackknife method to statistics, just half of the randomly selected data will be eliminated, rather than all of the observations. With this approach, the variance of the jackknife estimates of the mean will be n^2/(n - 1)^3 times smaller than that of the corresponding bootstrap estimates. [7] To sum up, at a data, an accurate model can be easily founded on computer with more collected samples, but since there are more samples come out, and then when these samples are stacked together, the result will be different compared to the beginning, although there is slightly different between each sample. Just like in mathematics, when 0.99 multiple each other, the final value will be different to the original value. Many probabilistic representation languages are limited in representing their richness, such as Bayesian networks, it's simple but unable to model variables. But now, more advanced languages are founded such as BUG, which also provides programming-language features including iteration and arrays, without being Turing complete. Now the languages can be modeled as long-running processes with many interacting entities and events. Now go back to soccer, for sports analytics, people can use accumulated statistics to make decisions. However, it's useless to just display data and make a conclusion. One thing can't be ignored. Data analysis should also be combined with the scene at that time, that is, the scene. This requires modeling many dependent events and interacting with players and teams. The same example also appears in the product launch. Same as the Goal, whether it can be successful is also uncertain. Data analysis can only help people in considering, but it's not determined. Some scenes, such as the alertness of competitors, cannot be statistically calculated.

As mentioned before, the Turing-Completed system is already existed, which is also called simulation language. Same as probabilistic programs, these simulations are randomly executed to produce different outputs. Furthermore, simulations can be widely used in public management, military planning, sports prediction, and other activities. However, the widespread use of sophisticated simulations demonstrates the need for rich probabilistic modeling languages. A probabilistic program is not limited to simulation, while simulation can only predict the future through probabilistic programming. Simulation can’t be used to Infer the cause of the result. Moreover, simulation is not able to simulate other important unknown information, that is, these factors cannot be detected. A probabilistic program is like a simulation, which provides its expectations when people make decisions, to help those making inferences. Since the founding of the probabilistic reasoning system, many new functions are being discovered. The Turing-Completed System is now getting useful in the society, such as Blockchain. The development of blockchain technology has the potential to support numerous codified agreements currently handled by conventional methods, including stock exchanges, monitoring contracts, land management records, food security, preserving provenance, and chain of custody maintenance. The infrastructure of daily life, such as business, social interaction, law, education, entertainment, nutrition, livelihood, housing, and so forth, will therefore incorporate technology [1-10]. Turing-complete systems are not perfect, at least now it still improving while it many defects are still exist. Maybe the data will be more accurate in the future. If it is accurate enough that people no longer need to use the brain, then the pros and cons will need to be weighted.

3. Summary

Probabilistic reasoning is everywhere, not just limited to a soccer game. On the way analyzing an event, it first infers the direction of the event based on the cause of the event, and then predict its outcome to form a complete model. And according to the principle of multiplication, although an initial model might be more accurate, as the number of samples increases, data start to dilute, then the final number obtained will be different to the original data(model). The main part of this article is to introduce Turing Completed Programming that have huge impact on people's daily lives. Back to the beginning of the article, when people facing something, and they are unable to see how the things go and don’t know to

DOI: 10.54254/2755-2721/6/20230816

deal with them, thus probabilistic programming appears with the development of society and help making references for them. With the development of technology, such programs that predict people's subsequent behavior directly based on the actual situations might be founded in the future, just like how the program behaves in the movie. But whether it can bring benefits more and how to deal with the disadvantages might face in the future, scientists will need more testification.

References

[1] Stigler, Stephen M. (1986). The History of Statistics: The Measurement of Uncertainty before 1 900. Cambridge: Harvard. ISBN 0-674-40340-1.

[2] Pfeffer, A.Practical probabilistic programming. Inductive logic programming (pp. 4-14). Spring er Berlin Heidelberg. https://doi.org/10.1007/978-3-642-21295-6_2

[3] De Raedt, L., & Kimmig, A. (2015). Probabilistic (logic) programming concepts. Machine Lear ning, 100(1), 5-47. https://doi.org/10.1007/s10994-015-5494-z

[4] Blitzstein, J. K., Hwang, J., & O'Reilly for Higher Education. (2015;2014;). Introduction to prob ability. CRC Press/Taylor & Francis Group. https://doi.org/10.1201/b17221

[5] Frank, T. D., SpringerLink ebooks - Physics and Astronomy, & Ebook Central. (2005;2006;). N onlinear fokker-planck equations: Fundamentals and applications. Springer.

[6] Rick Durrett. Probability Theory and Examples. Fourth Edition.

[7] ZouYang, WuHecheng, Zhao Yingding, Jiang Yunzhi. Random walk recommendation algorith m based on multi-weight similarity [J]. Computer Application Research, 2020,37(11):3267-3 270+3296.DOI:10.19734/j.issn. 1001-3695.2019.08.0275.

[8] Liao Yongxin. Research on the Joint Classification Algorithm of Heterogeneous Label Sets Bas ed on Random Walk and Dynamic Label Propagation [D]. South China University of Techno logy, 2017.

[9] Lu Yuke. Research on Complex Code Recognition Technology [D]. University of Electronic Sci ence and Technology of China, 2022. DOI: 10.27005/d.cnki.gdzku.2022.003580.

[10] Song Bin, Liu Lili, Zhang Lei, Wang Lei, Du Yuxin, Zhang Ning. Information Management of Coal Mine Equipment Based on Data Matrix Code [J]. Industrial and Mining Automation, 20 20, 46(11): 83-86+94. DOI: 10.13272/ j.issn.1671-251x.2020050059.